

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently amended) A method for handling failover of a data management application for a shared disk file system in a distributed computing environment having a cluster of loosely coupled nodes which provide services, comprising the steps of:

defining certain nodes of the cluster as failover candidate nodes;

storing configuration information for all the failover candidate nodes;

distributing message information including but not limited to failure information of at least one failover candidate node among the failover candidate nodes;

analyzing the distributed message information and the stored configuration information in order to determine whether to take over the service of a failure node by a failover candidate node or not;

in a first failover candidate node, taking over only a subset of the file system from the failure node;

carrying over a remaining portion of the file system not taken over by the first failover candidate node to at least a second failover candidate node; and

updating the configuration information in case of ~~at least one~~ the first failover candidate node and at least the second failover candidate node taking over the service of a failure node.

2. (Cancelled)

3. (Previously presented) The method according to claim 1, wherein the configuration information is stored in a central data storage arranged within the cluster.

4. (Previously presented) The method according to claim 1, wherein the distributed message information includes a failure report of at least one node.

5. (Previously presented) The method according to claim 1, wherein the failover candidate nodes calculate a priority key related to the workload of each of the failover candidate nodes which is distributed as part of the distributed message information.

6. (Previously presented) The method according to claim 5, wherein the failover candidate nodes receiving the priority key compare the received priority key with

their own priority key whereby the best priority key wins the right to take over the service.

7. (Previously presented) The method according to claim 1, wherein the updating of the configuration information is handled by means of a locking mechanism.

8. (Currently amended) An article of manufacture comprising a computer usable medium having computer readable program code means embodied therein for causing handling of failover of a data management application for a shared disk file system in a distributed computing environment having a cluster of loosely coupled nodes which provide services, the computer readable program code means in the article of manufacture comprising computer readable program code means for causing a computer to effect:

defining certain nodes of the cluster as failover candidate nodes;

storing configuration information for all the failover candidate nodes;

distributing message information including but not limited to failure information of at least one failover candidate node among the failover candidate nodes;

analyzing the distributed message information and the stored configuration information in order to determine whether to take over the service of a failure node by a failover candidate node or not;

in a first failover candidate node, taking over only a subset of the file system from the failure node;

carrying over a remaining portion of the file system not taken over by the first failover candidate node to at least a second failover candidate node; and

updating the configuration information in case of ~~at least one~~ the first failover candidate node and at least the second failover candidate node taking over the service of a the failure node.

9. (Currently amended) A system for handling failover of a data management application for a shared disk file system in a distributed computing environment having a cluster of loosely coupled nodes which provide services, comprising

data storage means for storing configuration information for failover candidate nodes;

communication interface means for distributing message information between the failover candidate nodes;

means for analyzing the message information and the configuration information in order to determine whether to take over the service of a failure node by a failover candidate node or not;

means for cascading the failover handling whereby a failover request is carried on to at least a second failover candidate node if only a subset of the file system is taken over from the failure node by a first failover candidate node; and

means for updating the configuration information in case of ~~at least one~~ the first failover candidate node and at least a second failover candidate node taking over the service of a failure node.

10. (Cancelled)

11. (Previously presented) The system according to claim 9, where the data storage means is a central data storage arranged within the cluster.

12. (Currently amended) The system according to claim 9, where the means for updating the configuration information are located at least at one of the first failover

candidate node and the at least second failover candidate node taking over a service of a
the failure node.